Downloaded from orbit.dtu.dk on: May 05, 2023

DTU Library

On the Impact of Layout Quality to Understanding UML Diagrams: Size Matters

Störrle, Harald

Published in:
Proceedings of 17th International Conference on Model Driven Engineering Languages and Systems

Publication date:
2014

Document Version
Peer reviewed version

Link back to DTU Orbit

Citation (APA):
Störrle, H. (2014). On the Impact of Layout Quality to Understanding UML Diagrams: Size Matters. In J. Dingel, W. Schulte, I. Ramos, S. Abrahao, & E. Insfran (Eds.), *Proceedings of 17th International Conference on Model Driven Engineering Languages and Systems* (pp. 518-534). Springer. Lecture Notes in Computer Science Vol. 8767

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

On the Impact of Layout Quality to Understanding UML Diagrams: Size Matters

Harald Störrle

Dept. of Applied Mathematics and Computer Science
Technical University of Denmark, hsto@dtu.dk

Abstract. Practical experience suggests that usage and understanding of UML diagrams is greatly affected by the quality of their layout. While existing research failed to provide conclusive evidence in support of this hypothesis, our own previous work provided substantial evidence to this effect. When studying different factors like diagram type and expertise level, it became apparent that diagram size plays an important role, too. Since we lack an adequate understanding of this notion, in this paper, we define diagram size metrics and study their impact to modeler performance. We find that there is a strong negative correlation between diagram size and modeler performance. Our results are highly significant. We utilize these results to derive a recommendation on diagram sizes that are optimal for model understanding.

1 Introduction

The Unified Modeling Language (UML) has been the "*lingua franca of software engineering*" for over a decade now. It is a generally held belief that visual languages are superior to textual languages in that they support human perceptual and thought processes, and that this is also true for the UML, in fact, that this is a major reason for the success of UML. However, there are actually few research results to support this belief. There *is* a large body of experimental results on the layout of UML class diagrams and how it affects human understanding and problem solving, but the findings are ambiguous, and sometimes unintuitive. In particular, only very small effects have been found in vitro. For instance, Eichelberger and Schmid note that "*We could not identify [...] a significant impact [by diagram quality].*" (cf. [9, p. 1696]).

On the other hand, practical experience in industrial software projects suggests a much higher impact of good or bad layout, and previous work by the author strongly supports this hypothesis (see [28, 29]). Inspection of our data and a qualitative study with our study participants suggested, however, that the size of the models portrayed in the diagrams might be a relevant factor. In order to study this question, we define a precise notion of diagram size and re-examine existing data sets of substantial size (78 participants, well over 1200 measurements). Our working hypothesis is that modeler performance correlates

negatively with diagram size. We also hypothesize, that layout quality[1] matters more with increasing diagram size: small diagrams are easy to use irrespective of the layout quality simply because they are small; modelers simply cope with bad layout. With increasing diagram size, however, the visual and/or mental capacity of a modeler is stretched, so that the layout quality impacts modeler performance. In other words, layout quality matters more, and is more apparent for larger diagrams. We analyze the diagram size metrics and various modeler performance indicators, including errors, preference/assessment, and cognitive load (cf. [13]).

If we can indeed correlate diagram size with modeler performance, however, we can exploit this relationship conversely to determine limits to the size of diagrams that afford being understood easily and correctly by modelers. Such limits might be helpful as guidelines to inexperienced modelers, such as students.

2 Related Work

The layout of graphs (in the mathematical sense) has been a longstanding research challenge, both with respect to automatic layout and to various aspects of usability, e.g., diagram comprehension, user preferences, and diagrammatic inference. Based on the rich knowledge on general graphs, research on the layout of UML has started with those of UML's notations that are closest to graphs, namely, class diagrams (cf. [23, 7, 10, 33, 18]), and, to a lesser extent, communication diagrams (see e.g. [17, 20] who use UML 1 terminology). Other types of UML diagrams, in contrast, have only attracted little interest so far (e.g. use case diagrams [8], or sequence diagrams, cf. [2, 32]). There is only little work on the Business Process Model and Notation (see [5]), and even less on UML activity diagrams [21].

Research on aspects of UML class diagrams has mostly focused on the impact of isolated low-level layout criteria such as line bends, crossings, and length. Unsurprisingly, each of these properties has little impact by themselves and are hard to prioritize. The more elusive higher levels like layout patterns, diagram flow, and the correspondence between a diagram and its intended message seem to have not yet been studied empirically at all. The influence of the expertise level, on the other hand, has been studied [1, 22].

The main focus of previous work on UML diagram types and their layout has been with one of four aspects: diagram comprehension (cf. [25, 26, 15, 20] and/or user preference (cf. [18, 31]), automatic layout (cf. [7, 10, 16, 8, 4]), or one of a variety of diagram inference tasks, e.g., program understanding based on visualizations (cf. [32]), or the role of design patterns in understanding (cf. [26, 27]).

Most research uses controlled experiments and evaluate user performance using paper questionnaires, or online surveys. Only a few contributions have used other methods, most notably eye tracking (see [3, 33, 26]). After using both

[1] We will elaborate on the notion of layout quality in Section 3 below.

methods for essentially the same experiment, Sharif et al. have concluded that these two methods are mostly complementary wrt. comprehension tasks (cf. [24]). Thus, eye tracking is only favorable for a tightly restricted set of research questions, in particular when taking into account the considerable cost and effort involved. Having said that, most questionnaire-based approaches employ only very few participants in their experiments, typically in the range of 15 to 30, with the notable exceptions of [25], [19] and [2] involving 45, 55 and 78 participants, respectively. The research done for the current paper involved 78 participants.

3 "Good" layout of UML diagrams

In this section, we will briefly review the knowledge on aesthetic criteria for the layout of UML diagrams and its effects on model understanding. A detailed discussion of aesthetic criteria for class diagrams is found in [7, p. 54–65], a recent survey of empirical results on layout criteria is found in [9]. Wong and Sun [32] provide an overview of these criteria from a cognitive psychology point of view, along with an evaluation of how well these principles are realized in several UML CASE tools. Purchase et al. discuss aesthetic criteria with a view to the layout of UML class and communication diagrams (cf. [18, 17]) and also provide sources to justify and explain these criteria (cf. [15]). Eichelberger [6] also discusses these criteria at length, and shows how they can be used in the automatic layout of UML class diagrams.

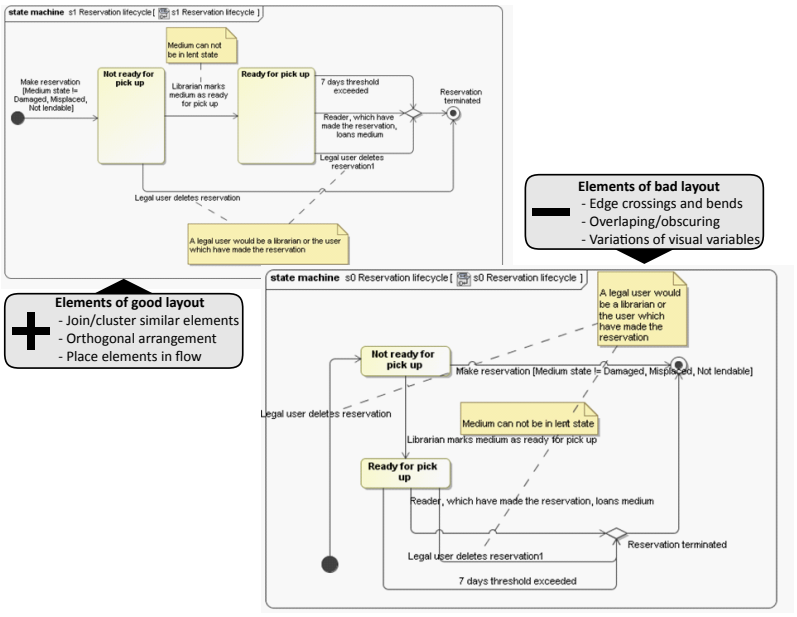

Fig. 1. Examples of good/bad layouts of a diagram as used in the study

The layout of UML diagrams is governed by four levels of design principles. First, there are the general principles of graphical design and visualization that apply to all kinds of diagrams, and probably any kind of visualization. For instance, in a good layout, elements should not obscure each other, the Gestalt principles should be respected [12], text should be shown in a readable size, elements should be aligned (e.g., on a grid), and there should be sparing and careful use of colors, and different fonts or styles.

Second, there are layout principles applying to all structures that can be considered as a graph, mathematically speaking. Thus, good layouts should avoid or minimize crossings, bends, and length of lines. Most of the empirical research on UML diagrams focuses on principles from this level, e.g., [23, 7, 10, 33, 18].

Third, there are layout principles that apply mostly only to notations like those found in UML. For instance, diagrams with some inherent ordering of elements should maintain and highlight that ordering as visual flow. Visual clutter should be reduced by introducing symmetry when possible. For instance, similar edges should be joined, similar elements should be aligned and grouped, and so on. In UML, this means that if a class has several subclasses, it might be helpful to group and align the subclasses and join the arcs indicating the inheritance-relationship. Another application is found in activity diagrams, where several consequences of a decision could be aligned and grouped.

Fourth, there is the level of pragmatics, that is, support for underlining the purpose of a diagram in order to better address the audience. Items may be highlighted by color, size, or position to guide and direct the attention of readers. On this level, rules and guidelines from lower levels may be put aside to better serve the paramount purpose of conveying the message and telling whatever story the diagram designer intends to tell.

In order to develop algorithms for creating automatic layouts that are perceived as being helpful (or "good") by human modelers, detailed knowledge about the individual criteria, their relative and absolute impact, and their formalization is needed. So, it is not surprising that most of the empirical research on UML diagrams has so far focused on studying individual principles, with an emphasis on the second group (cf. [23, 7, 10, 33, 18]). For instance, work by Purchase et al. has shown that there are many such criteria with varying degrees of impact (see e.g. [18]), though all of them seem to have a rather small impact with findings that are not highly or not at all statistically significant. Also, the ranking and contribution of these criteria may vary across different diagram types. Even between class and communication diagrams, which are rather close relatives as far as concrete syntax is concerned, [18, pp. 246] shows notable differences in the ordering and impact of layout criteria. Thus, other notations that share even less commonalities with class diagrams (e.g., activity, use case, or sequence diagrams) may need a completely different set of criteria.

For humans creating diagram layouts, on the other hand, a set of comparatively vague guidelines together with some instruction is often good enough for practical purposes. Humans may (and will) mix and match criteria from all four levels as appropriate and create what they *and their peers* perceive as high qual-

ity UML diagrams. Of course, there is still a large degree of subjectivity in this definition, but it does capture the intuition (see [28, 29] for detailed evidence). Therefore, in the remainder of this paper, we will call a diagram (layout) *good*, if it mostly adheres to the criteria from all these levels, and *bad* if it mostly violates them. Generally speaking, in terms of the four levels of layout rules described above, if a diagram layout does not (significantly) violate any of the rules on the first two levels but (more or less) adopts the rules described in the latter two levels we call it a "good" layout. Conversely, we call a diagram layout "bad" if it consistently violates these rules.

4 Size of UML diagrams

Surprisingly, there seems to be no metric for the size of UML diagrams that we can use as a basis in our correlation. So, we have to define such a metric. We will visit three of them to find the most appropriate, starting with the simplest conceivable approach of simply counting the number of diagram elements (not to be confused with the number of model elements presented or implied in the diagram). This metric has the advantage of being straightforward to compute, but does not take into account differences among the potential elements of a diagram.

Arguably, this metric is not just simple, but too simple, as it implies that all diagram elements contribute the same amount of complexity and information to the diagram. If this assumption does not hold true, we could introduce a weight factor for the individual types of elements to compensate for differences between different element types. It is not quite clear, however, what the "right" weights should be, and how to obtain them.

As a pragmatic approach to defining weights uniformly, we use the approach pursued in [30], and provide a simple classification of the elements of UML diagrams into lines, shapes, and labels, and assign one of three complexity levels to each of them (simple, medium, and large), according to the amount of cognitive load we may expect involved in processing them based on the laws of Gestalt psychology (cf. [12]). For instance, a plain association might be considered a simple line, an association with an adornment on one side (such as a composition or a directed association) might be considered a line of medium complexity, and an association with adornments on both sides might be considered a line with large complexity. Lines with several legs could be understood as sets of lines, e.g., an association with two adornments and two corners decomposes into one simple line and two medium lines (one adornment each).

- Lines include all kinds of straight or curved lines. Lines made up of n different segments are considered as n different lines. Decorations at the beginning or end of a line or line segment (such as arrow heads) are considered to be an integral part of the line but increase its complexity.
- Shapes include the basic geometric shapes like circles, rectangles, and ellipses as simple elements. Shapes that occupy a large area of a diagram and contain

other shapes are considered to be of medium complexity, while the more complex iconic shapes like a stick-person or a lightning-arrow are considered complex shapes.

– Labels are strings of text that are attached to or positioned relative to other elements. Labels are restricted to single lines. Single characters or short names are considered simple, long names are considered as medium complex, and structured expressions like sentences or operation declarations are considered to be highly complex.

With these conventions, we define diagram size as the number of elements in a diagram, weighted by their complexity (e.g., one might define the weights S: 1, M: 1.5, L: 2). This metric is substantially more difficult to compute than our first proposal above, but it reflects the intuition more accurately, and could thus be expected to be more realistic, and provide higher validity.

Still, one might argue that the second approach is too simplistic, as the influence of diagram types is not considered. After all, every UML diagram establishes a context that restricts the admissible vocabulary in this diagram to a small subset that is available for the given diagram type. For instance, there are many more notational elements in the UML sub-language of Activity Diagrams than there are in the sub-language of Use Case Diagrams. Thus, according to classic information theory, the weight of any element in an Activity Diagram ought to be higher than the weight of the elements in Use Case Diagrams.

In analogy with classic information theory, the number of choices should determine the information content (i.e., the weight) of a diagram element. We compute the information content of diagram elements as the binary logarithm of the set of similar elements a modeler may chose from, per diagram type. So, for every diagram element e from a class E of diagram elements in a given diagram type, we compute $\mathit{weight}(e) = \mathit{log}_2(|E|)$. Using this as a weight factor provides a third metric of diagram size.

Note that we disregard topological information (i.e., containment). Thus, our metric is not necessarily a measure of diagram complexity or information content. Clearly, we will need to validate these diagram size metrics. So, we computed the sizes according to each measure with some (sensible) variations for the weights of the second metric for the same 38 diagrams that have been used in [28, 29]. We compared the outcomes using Pearson's product-moment correlation. Surprisingly, we found that all three measures show a very high level of correlation (between 0.967 and 0.992) with very high confidence ($p < 10^{-15}$). That is to say: the measures do not yield significantly different results. In other words, it does not matter which metric we use. So, we decide for the one that offers the practical advantage of being simple to compute, that is, in the remainder we simply count the number of diagram elements as a metric for diagram size.

5 Experimental setup

We used [14] as a guideline for our experimental setup. We presented the participants with paper questionnaires showing one UML diagram and ten questions on the diagram, recording four categories of answers (right, wrong, "don't know", and no answer), time used, subjective assessment of the task difficulty (three questions in experiment D, and four questions in experiments E and F). The questionnaires also contained a separate sheet where we asked for personal preference, and subjective assessment of layout quality. The dependent variables are accuracy and speed of comprehension, and preference. The independent variables are the experience level of the participants (beginner/advanced/elite), the diagram type (class, sequence, state machine), the diagram size (small/large), and, of course, the layout quality (good/bad). Altogether, we ran three experiments with together 78 participants, and a completion rate of 80%. Minor adjustments and corrections were made as compared to the experimental setup reported in [28]. In the remainder, we will focus on the setup of the second and third experiment. The details of the setup are discussed below; a summary of the experimental setup and study design is shown in Fig. 2.

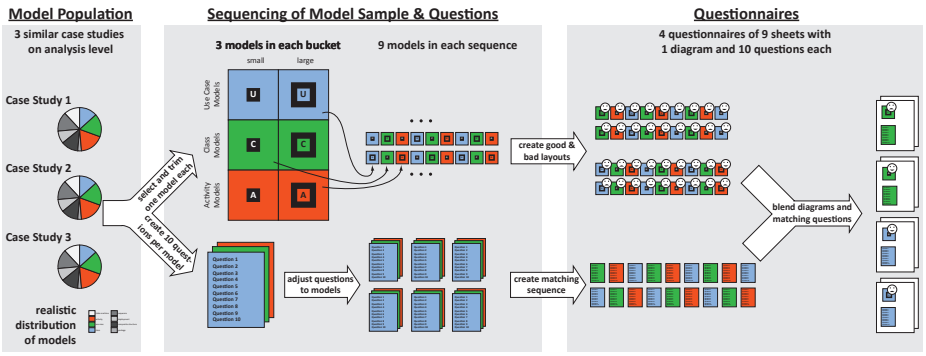

Fig. 2. The experimental setup and study design

5.1 Model population

The models used in the experiments have been created by students as part of their coursework in a requirements engineering course taught by the author. These models belonged to one of three case studies and have been prepared by teams of 4-7 students over a period of twelve weeks with an approximate effort of 600-800 working hours for each model. For each case study, two or three teams worked in parallel; for each case study, the model of the team achieving the highest grade was selected. This procedure ensured several desirable properties.

Firstly, by using models created by students undergoing the same course and being awarded the same grade, very similar levels of modeler capability and model quality may be assumed. Furthermore, the models used exhibit a large degree of methodological homogeneity in that they are very similar in terms of model structure and size, model and diagram usage, and frequency distribution of diagram types. Also, in the models used in our experiments, model elements had their original, semantic-bearing names, whereas in some previous experiments this vital aspect seems to have been deliberately eliminated by giving meaningless synthetic names to model elements (cf. [9, p. 1697]). Secondly, due to the project oriented nature of the course, we can assert that the models underlying our experiment are realistic in the sense that their size, quality, and purpose are very close to industrial reality. Finally, all of these models used exist at the same stage of the software life cycle, namely requirements analysis.

In contrast, all earlier works seem to have used only a single case study and model, and most work has been carried out on models at the design or implementation level. Also, there is no indication in previous work as to how close to the reality of practical software development the underlying models are.

5.2 Diagram samples and questions

From each of the three model types selected from the model population, we chose one large and one small example of class, state machine, and interaction diagrams with particularly good or bad layout. The quality of layout is measured by the adherence or non-adherence to a number of layout rules discussed in the related work (see Section 3). This step yielded three models (one from each case study) for each of the six buckets, that is, the categories of small/large diagrams of types class/activity/use case. So we arrived at 18 models altogether which were then trimmed to fit onto a questionnaire page. We then derived two variants from each diagram exhibiting good and bad layout (i.e., two different treatments), respectively, yielding 36 different diagrams (see Fig. 1 above for good/bad layouts of a diagram; a sample questionnaire can be found at `www2.compute.dtu.dk/~hsto/downloads/q2.pdf`).

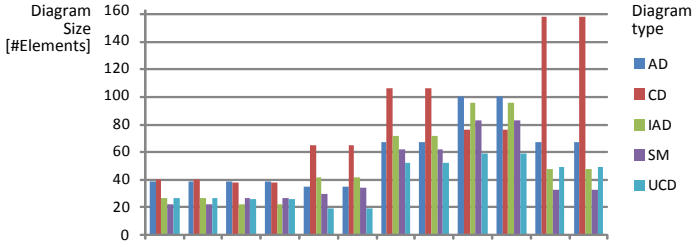

Fig. 3. Distribution of sizes and types of the diagrams used in the experiments.

5.3 Participants and completion rates

The participants for experiments 1 and 2 were recruited among students from different computer science classes at the Danish Technical University in Lyngby. The participants for experiment 3 were recruited among elite graduate students and staff from the University of Augsburg.[2] All participants took part voluntarily with no reward or threat and under complete anonymity, i.e., it was clear to students that their performance had no influence whatsoever on their grades, for instance. Immediately before the experiment, all participants received a ten-minute introduction to those parts of the UML that were covered in the experiment.

The participants showed a wide spread in UML knowledge. In all experiments, in the core parts of the questionnaire, nine diagrams were presented and ten questions were asked per diagram. We saw an overall completion rate of these core questions of over 80%. See Table 1 for more details on the population.

Table 1. Demographic data on the participants of all experiments, "completion" refers to the completion rate on core questions

Experiment	male	female	all	completion
1 (BEng)	29	3	33	75.1%
2 (MSc)	29	5	34	82.6%
3 (Elite)	10	1	11	90.1%
all	**68**	**9**	**78**	**82.6%**

6 Results

6.1 Correlations between diagram size and modeler performance

As outlined above, our initial hypothesis was that there is a correlation between diagram size and modeler performance in understanding these diagrams. Plotting the diagram size as defined above against the performance on all diagrams yielded the scatter plots shown in Fig. 4. Adding trend-lines reveals that the correlation is indeed present: with increasing diagram size, the mean score decreases while the variance increases. Similarly, perceived diagram clarity decreases with increasing diagram size. Surprisingly, there is also a positive correlation between diagram size and perception of layout quality.

We then tested properly for correlations between diagram size and modeler performance. We used the simple diagram size metric, as discussed above,

[2] These experiments correspond to the experiments D, E, and F reported before in [29].

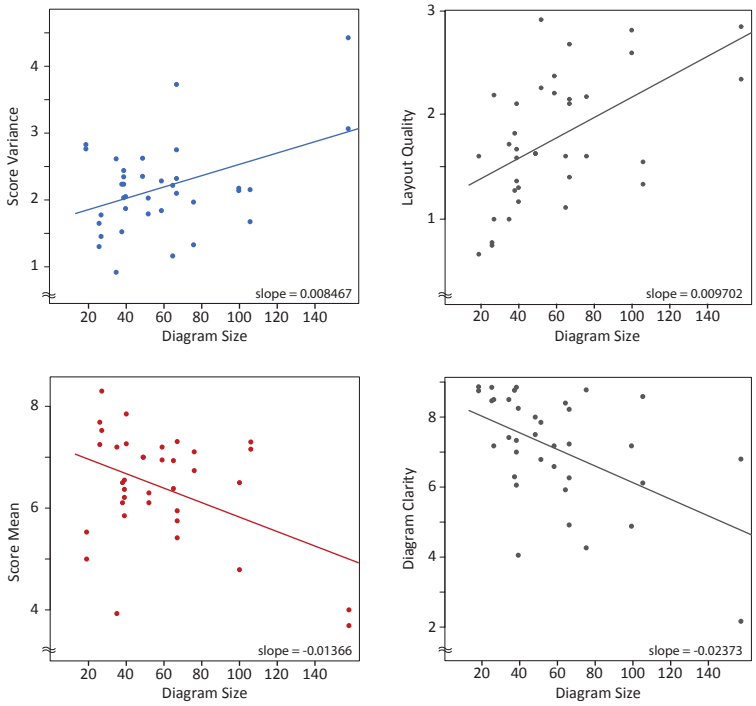

Fig. 4. Plots of various measures of modeler performance against diagram size (clockwise from left bottom): score mean, score variance, subjective assessment of layout quality and diagram clarity. The trend-lines are created from linear models.

and correlated it with all measures of modeler performance observed in our experiments. We calculated the correlations between diagram size and modeler performance using Pearson's product-moment correlation (`cor.test` in R). We assess the effect size of a correlation of up to 0.3 to as small (S), as large (L) for values over 0.4, and as medium (M) for values in between, see Table 2.

It is quite clear that there is indeed a large correlation between increasing diagram size and decreasing mean scores. This is in line with the observation that the variance increases with diagram size: increased difficulty will provoke a greater spread of results. We have seen a similar effect in our previous studies, where the natural variance in capability of the population becomes more visible when testing poor layouts because these help less with diagram understanding. For the good layouts, individual performance differences matter less, as they are, partially, leveled by the helpful layout. This objective measure is further confirmed by the subjective measure of asking the participants to assess the clarity of the diagrams: uniformly large correlations are found between increasing diagram size and decreasing clarity. Yet more confirmation is found when considering the subjective assessment of cognitive load: with increasing size, cognitive load as

Table 2. Pearson's product-moment correlation between diagram size and modeler performance, measured as mean and variance of objective performance (correct answers, i.e., score), different subjective assessments, and cognitive load measures. In each cell, the first number is Pearsons's r indicating the size of the correlation, the letter S/M/L classifies the effect size, the next number is the p-value, and the stars indicate its significance level.

Objective Performance	**Score Mean**			**Score Variance**		
	r	ES	p SIG	r	ES	p SIG
All Diagrams	−0.423	L	0.010 **	0.424	L	0.010 **
Bad Layout	−0.491	L	0.039 *	0.534	L	0.023 *
Good Layout	−0.396	M	0.104 *	0.303	M	0.222

Diagram Assessment	**Layout Quality**			**Layout Clarity**		
	r	ES	p SIG	r	ES	p SIG
All Diagrams	0.538	L	< 0.001 ***	−0.508	L	0.002 **
Bad Layout	0.521	L	0.027 *	−0.563	L	0.015 *
Good Layout	0.573	L	0.013 *	−0.766	L	0.0002 ***

Cognitive Load	**Diagram Understanding**			**Diagram Complexity**		
	r	ES	p SIG	r	ES	p SIG
All Diagrams	−0.338	M	0.044 **	−0.081	S	0.640
Bad Layout	−0.452	L	0.060 *	−0.313	M	0.207
Good Layout	−0.197	S	0.434	0.152	S	0.548

expressed by subjective assessment of task complexity increases, too. Observe that subjective assessment has been found to be highly correlated with objective measures of cognitive load [11], and that both questions asked to measure cognitive load exhibit similar patterns. The negative correlation between diagram size and perceived diagram complexity might be an experimental artifact since it has no statistic significance and relatively small effect sizes.

We also see a positive correlation between diagram size and layout quality, which seems to contradict our hypothesis. We explain this by observing that it is literally obvious to most modelers that a diagram has high quality when presented with one. Answering this question for a poor diagram, on the other hand, is much harder, as it requires knowledge about what makes a poor diagram, too. In particular novice modelers have yet to appreciate the negative impact of line crossings, bends, obscuring elements and so forth.

All of these effects are substantially stronger for poor layouts than for good layouts. This is in support of our initial hypothesis that layout quality matters more with increasing diagram size. In other words: small diagrams are easy to use anyway, so bad layout can be easily compensated. For larger diagrams, however, when the visual and/or mental capacity of a modeler is reached or exceeded, the impact of layout quality becomes visible: layout quality matters more, and is more apparent for larger diagrams.

The results for objective measures and subjective assessments seem to provide stronger results than the results for cognitive load measures, although this might be attributable to factors outside of the experimental control.

6.2 Correlations differentiated by expertise level

Table 3. Pearson's product-moment correlation between diagram size and modeler performance, controlled for expertise level.

Objective Performance	**Score Mean (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	−0.494	L	0.002	**	0.018	S	0.917	
Bad Layout	−0.397	M	0.103	.	−0.173	S	0.493	
Good Layout	−0.615	L	0.007	**	0.243	M	0.331	

Objective Score	**Score Variance (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	0.290	M	0.086	.	0.053	S	0.764	
Bad Layout	0.254	M	0.309		0.204	M	0.432	
Good Layout	0.343	M	0.163		−0.085	S	0.736	

Diagram Assessment	**Layout Quality (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	0.569	L	0.0003	***	0.484	L	0.003	**
Bad Layout	0.534	L	0.023	*	0.516	L	0.028	*
Good Layout	0.615	L	0.007	**	0.536	L	0.022	*

Diagram Assessment	**Layout Clarity (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	−0.525	L	0.001	***	−0.440	L	0.007	**
Bad Layout	−0.742	L	0.0004	***	−0.698	L	0.001	**
Good Layout	−0.554	L	0.017	*	−0.570	L	0.014	*

Cognitive Load	**Diagram Understanding (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	−0.313	M	0.063	.	−0.199	S	0.245	
Bad Layout	−0.184	S	0.465		−0.064	S	0.800	
Good Layout	−0.421	L	0.082	.	−0.306	M	0.218	

Cognitive Load	**Diagram Complexity (low/high expertise)**					
	r	ES	p	SIG	r	ES	p	SIG
All Diagrams	−0.082	S	0.634		0.042	S	0.808	
Bad Layout	0.133	S	0.600		0.251	M	0.315	
Good Layout	−0.349	M	0.156		−0.134	S	0.595	

Previous work by Abraho, Ricca and others [1, 22] suggests that the expertise level is important in diagram understanding, and when controlling for expertise levels, more interesting phenomena become visible (see Table 3). In this table, we have used the same arrangement of values in cells as in Table 2, but have split the data between modelers with lower and higher levels of expertise (left and right, respectively). First of all, let us establish that there is indeed a performance difference in expertise level in our sub-populations. Using a one-sided Wilcoxon-test to compare the average score on good layouts for the two sub-populations, we can reject the hypothesis that the sub-populations exhibit the same performance with very high significance ($p = 0.00013$). When comparing the scores, score variances, and the cognitive load measures, participants with high expertise level are much less affected by increasing diagram size than participants with lower expertise levels. This holds irrespective of layout quality, but is even stronger for poor layouts. Some of these findings are not statistically significant, however, since analyzing the sub-populations separately drastically decreases the number of data points. Still, all correlation show the same pattern and tendencies which does add evidence to our earlier observations.

Even with the reduced population size we find significant or highly significant correlations between increasing diagram size and reduced layout clarity, particularly for poor layout where correlation exceeds -0.7 ($p < 10^{-3}$). Again, the effect is larger for poor layouts than for good ones, and again, the same pattern is found in the cognitive load measures ("Understanding" and "Complexity"), though the latter findings are not statistically significant.

6.3 Optimal diagram size

Based on our data, we can compute trend-lines of the correlations, as shown in Fig. 4 (bottom right). Computing a linear model yields coefficients of a linear equation ($intercept = 7.21, slope = -0.014$). This allows us to compute the diagram sizes at which the study participants answered a given number of questions about the diagrams correctly. It seems natural to use the boundaries of the second and third quartile as lower bound, optimum, and upper bound of expected performance. The values for these boundaries and a geometric interpretation of the relationship between quantiles of score and optimal size is given in Fig. 5.

In practice, the quality of diagrams and modelers will vary widely. When disregarding these factors, we conclude that diagrams with approximately 20 to 60 diagram elements should allow average modelers to answer approximately half of the questions about the model represented by the diagram correctly. Thus, an objective recommendation for boundaries of diagram size would be in this range, too. It would be trivial to implement such a function in a modeling tool, which could provide guidance to modelers.

7 Threats to validity

Internal validity Great care has been taken to provide systematic permutations of diagrams, questions, and sequences thereof to avoid bias by carry-over ef-

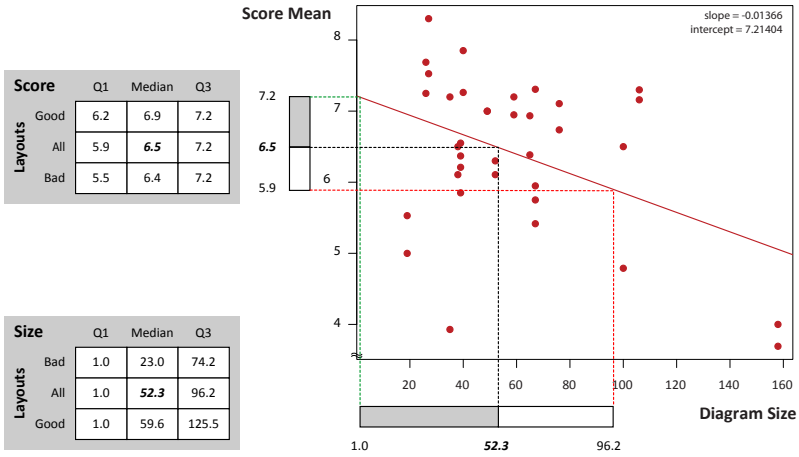

Fig. 5. The red trend-line visualizes the correlation between scores and diagram sizes. Geometrically speaking, this means to mirror the distribution of scores at the `size score` trend-line. Observe that high scores correlate to small diagram sizes.

fects ("learning"). Any such effects would occur similarly for all treatments and, thus, would cancel each other out. Participants have been assigned to tasks randomly. We can also safely exclude bias through the experimenter himself, since there were only written instructions that apply to all conditions identically. We correlated it with different measures, each of which was measured in multiple different ways to reduce the danger of introducing bias through the experimental procedure.

External validity The selection of the models and diagrams may be a source of bias. However, we applied objective and rational criteria to the selection, and compared to previous similar studies, we used three different diagram types (rather than just one or two), a competitively large number of models, and very realistic models. The layouts for the models were, to a large degree, used-as-found, that is, they were created under realistic conditions by people unconnected to these experiments. On top of that, our study is based on a comparatively large number of participants. So, the present study is certainly among the best validated among studies of its kind and we expect our results to be valid for UML models *in general*, i.e., we expect a markedly higher degree of external validity than previous contributions can claim.

Conclusion validity We have used non-parametric tests, where applicable, to compensate for skewed distributions in our data. We have consistently provided statistical significance level and the effect size with our inferences. Due to the (relatively) high number of study participants, most of the inferences we present are equipped with high or very high levels of statistic significance and large effect sizes, using Cohn's thresholds for the effect size levels for want of any better

guideline. When controlling for sub-populations, the significance levels decrease, but keep showing the same patterns which is sufficient for the claims we make based on these data. We do assume a linear correlation between variables prima facie, but this is justified by an earlier ANOVA-analysis where the squared terms were much too small to have a significant impact on our study.

Construct validity Gopher and Braune [11] show that subjective assessments of cognitive load is accurate in the sense that it correlates strongly with objective measures such as skin conductivity, pupillary response, or heart rate. Categorizing layout quality as good and bad was done based on existing findings on layout understanding and aesthetics (see Section 2 for more details), which in turn are grounded in the well-established findings of Gestalt psychology.

There is no established metric for "diagram size" in the context of UML or similar notations. We have developed different metrics but found that they all correlate highly. Thus, we have opportunistically adopted the simplest of these metrics. There is no particular evaluation as to whether this construct is valid.

8 Conclusion

In earlier work, we established that layout quality does impact the understanding of UML diagrams [28], and that this applies irrespective of diagram type, but dependent on modeler expertise [29]. We could so far not answer the question whether diagram size had an influence, and, if so, what its magnitude would be. Thus, in this paper, we developed measures for the size of UML diagrams. Since they correlate almost perfectly on a population of 38 diagrams, we concluded that it is irrelevant which of these diagram size metrics is used. Thus we chose the pragmatically simplest metric.

Using this diagram size metric, we re-analyzed existing data sets and find strong evidence in support of our hypothesis. We conclude that high layout quality is particularly helpful for large diagrams, and that it is particularly helpful for modelers with low expertise. Based on these findings, we derive pragmatic guidelines on the optimal size of diagrams that are very easy to apply in tools, based on objective findings, and promise to be beneficial to many modelers.

The experimental procedure has been designed carefully to exclude bias of any kind, learning effects, and distortion. We have included a relatively large number of participants ($n = 78$) in our experiments, as a further contribution to validity. Most of the tests and correlations we have computed are equipped with high or very high levels of statistical significance. We consistently report completion rates, effect sizes, and similar data to allow scrutinizing our results, and allow other scientists to conduct secondary research based on our work. Thus we conclude, that our findings have a high level of validity.

Consistent with previous findings reported in [28, 29], a stronger effect is seen in subjective measures (cognitive load, assessment) than in objective measures (score), pointing to cognitive mechanisms to cope with diagram complexity. We hypothesize that increasing extrinsic cognitive load will lead to stronger effects

in the objective measures. One way of doing this is through dual-stimulus experiments.

References

1. Silvia Abrahão, Carmine Gravino, Emilio Insfrn, Giuseppe Scanniello, and Genoveffa Tortora. Assessing the Effectiveness of Sequence Diagrams in the Comprehension of Functional Requirements: Results from a Family of Five Experiments. *IEEE Txn. SE*, 39(3):327–342, 2013.
2. Carol Britton, Maria Kutar, Sue Anthony, Trevor Barker, Sarah Beecham, and Vitoria Wilkinson. An empirical study of user preference and performance with UML diagrams. In *Proc. IEEE 2002 Symp. Human Centric Computing Languages and Environments (HCC/LE)*, pages 31–33. IEEE, 2002.
3. Shehnaaz Y.P. Dawoodi. *Assessing the Comprehension of UML Class Diagrams via Eye Tracking*. PhD thesis, Kent State University, 2007.
4. Tim Dwyer, Bongshin Lee, Danyel Fisher, Kori I. Quinn, Petra Isenberg, George Robertson, and Chris North. A Comparison of User-Generated and Automatic Graph Layouts. *IEEE Txn. Visualization and Computer Graphics*, 15(6):961–968, 2009.
5. Philip Effinger, N. Jogsch, and S. Seiz. On a Study of Layout Aesthetics for Business Process Models Using BPMN. In *Proc. 2nd Intl. Ws. Business Process Modeling Notation (BPMN)*, pages 31–45. Springer Verlag, 2010.
6. Holger Eichelberger. Aesthetics of class diagrams. In *Proc. 1st Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*, pages 23–31. IEEE, 2002.
7. Holger Eichelberger. *Aesthetics and automatic layout of UML class diagrams*. PhD thesis, University of Würzburg, 2005.
8. Holger Eichelberger. Automatic layout of UML use case diagrams. In *Proc. 4th ACM Symp. Software Visualization (SOFTVIS)*, pages 105–114. ACM, 2008.
9. Holger Eichelberger and K. Schmid. Guidelines on the aesthetic quality of UML class diagrams. *Information and Software Technology*, 51(12):1686–1698, 2009.
10. Markus Eiglsperger. *Automatic layout of UML class diagrams: a topology-shape-metrics approach*. PhD thesis, Universität Tübingen, 2003.
11. Daniel Gopher and Rolf Braune. On the Psychophysics of Workload: Why Bother with Subjective Measures? *Human Factors*, 26(5):519–532, 1984.
12. Kurt Koffka. *Principles of Gestalt Psychology*. Routledge & Kegan Paul, 1935.
13. Fred Paas, Juhani E. Tuovinen, Huib Tabbers, and Pascal W.M. Van Gerven. Cognitive Load Measurement as a Means to Advance Cognitive Load Theory. *Educational Psychologist*, 38(1):63–71, 2003.
14. Shari Lawrence Pfleeger. Experimental design and analysis in software engineering. *Annals of Software Engineering*, 1(1):219–253, 1995.
15. H.C. Purchase, L. Colpoys, D.A. Carrington, and M. McGill. *UML Class Diagrams: An Emprical Study of Comprehension*, pages 149–178. Kluwer, 2003.
16. Helen C. Purchase. Metrics for Graph Drawing Aesthtetics. *J. Visual Languages and Computing*, 13(5):501–516, 2002.
17. Helen C. Purchase, Jo-Anne Allder, and David A. Carrington. Graph layout aesthetics in UML diagrams: user preferences. *J. Graph Algorithms Applications*, 6(3):255–279, 2002.
18. Helen C. Purchase, David Carrington, and Jo-Anne Allder. Empirical Evaluation of Aesthetics-based Graph Layout. *J. Empirical Software Engineering*, 7(3):233–255, 2002.

19. Helen C. Purchase, David A. Carrington, and Jo-Anne Allder. Experimenting with aesthetics-based graph layout. In M. Anderson, P. Cheng, and Volker Haarslev, editors, *Proc. Intl. Conf. Theory and Application of Diagrams (Diagrams)*, number 1889 in LNAI, pages 489–501. Springer Verlag, 2000.
20. Helen C. Purchase, Linda Colpoys, Matthew McGill, and David Carrington. UML Collaboration Diagram Syntax: An Empirical Study of Comprehension. In *Proc. 1st Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*, pages 13–22. IEEE Computer Society, 2002.
21. Gianna Reggio, Filippo Ricca, Giuseppe Scanniello, Francesco Di Cerbo, and Gabriella Dodero. On the comprehension of workflows modeled with a precise style: results from a family of controlled experiments. *Software & Systems Modeling*, pages 1–24, 2013.
22. Filippo Ricca, Massimiliano Di Penta, Marco Torchiano, Paolo Tonella, and Mariano Ceccato. How Developers' Experience and Ability Influence Web Application Comprehension Tasks Supported by UML Stereotypes: A Series of Four Experiments. *IEEE Txn. SE*, 36(1):96–118, 2010.
23. J. Seemann. Extending the Sugiyama algorithm for drawing UML class diagrams: Towards automatic layout of object-oriented software diagrams. In *Proc. Intl. Conf. Graph Drawing (GD)*, pages 415–424. Springer, 1997.
24. Bonita Sharif and Jonathan I. Maletic. An empirical study on the comprehension of stereotyped UML class diagram layouts. In *Proc. 17th IEEE Intl. Conf. Program Comprehension (ICPC)*, pages 268–272. IEEE, 2009.
25. Bonita Sharif and Jonathan I. Maletic. The effect of layout on the comprehension of UML class diagrams: A controlled experiment. In *Proc. 5th IEEE Intl. Ws. Visualizing Software for Understanding and Analysis (VISSOFT)*, pages 11–18. IEEE, 2009.
26. Bonita Sharif and Jonathan I. Maletic. An eye tracking study on the effects of layout in understanding the role of design patterns. In *Proc. 2010 IEEE Intl. Conf. Software Maintenance (ICSM)*, pages 41–48. IEEE, 2010.
27. Bonita Sharif and Jonathan I. Maletic. The Effects of Layout on Detecting the Role of Design Patterns. In *Proc. 23rd IEEE Conf. Software Engineering Education and Training (CSEE&T)*, pages 41–48. IEEE, 2010.
28. Harald Störrle. On the Impact of Layout Quality to Unterstanding UML Diagrams. In *Proc. IEEE Symp. Visual Languages and Human-Centric Computing (VL/HCC'11)*, pages 135–142. IEEE Computer Society, 2011.
29. Harald Störrle. On the Impact of Layout Quality to Unterstanding UML Diagrams: Diagram Type and Expertise. In G. Costagliola, A. Ko, A. Cypher, J. Nichols, C. Scaffidi, C. Kelleher, and B. Myers, editors, *Proc. IEEE Symp. Visual Languages and Human-Centric Computing (VL/HCC'12)*, pages 195–202. IEEE Computer Society, 2012.
30. Harald Störrle and Andrew Fish. Towards an Operationalization of the "Physics of Notations" for the Analysis of Visual Languages. In Ana et al. Moreira, editor, *16th Intl. Conf. Model Driven Engineering Languages and Systems (MoDELS'13)*, number 8107 in LNCS, pages 104–120. Springer Verlag, 2013.
31. Jennifer Swan, Maria Kutar, Trevor Barker, and Carol Britton. User Preference and Performance with UML Interaction Diagrams. In *Proc. 2004 IEEE Symp. Visual Languages and Human Centric Computing (VL/HCC)*, pages 243–250. IEEE, 2004.
32. Kenny Wong and Dabo Sun. On evaluating the layout of UML diagrams for program comprehension. *Software Quality Journal*, 14(3):233–259, 2006.

33. Shehnaaz Yusuf, Huzefa Kagdi, and Jonathan I. Maletic. Assessing the Comprehension of UML Class Diagrams via Eye Tracking. In *15th IEEE Intl. Conf. Program Comprehension (ICPC'07)*, pages 113–122. IEEE Computer Society, 2007.